



Repetir código no es de buenos programadores

Transcripción

[00:00] Bueno, amigos y amigas, vamos a continuar con nuestro curso de lógica de programación. Vamos a reutilizar el código que habíamos hecho en el anterior aula, donde lo que hicimos fue graficar y personalizar la bandera de Italia, en este caso, utilizamos algunas funciones de Canvas, utilizando la etiqueta de Canvas y usando funciones para graficar figuras geométricas: rectángulos, cuadrados, circunferencias.

[00:38] Vamos a continuar aprendiendo, reutilizando estos conceptos de gráficos y vamos a asociar reutilizar los conceptos que habíamos visto en las dos anteriores partes de este curso de lógica de programación: funciones, condiciones, loop y todos estos conceptos que son la base de un programador.

[01:04] Entonces, vamos a reutilizar este programa, este código, solo que voy a crear uno nuevo. Aquí programa 2. Entonces, lo que voy a reutilizar en realidad es casi todo. Voy a solo eliminar esto.

[01:28] Aquí estoy reutilizando la etiqueta Canvas, solo recordando lo que hacía aquí cada uno de los comandos, en el cual, definíamos nuestra pizarra, recordando que Canvas es como si fuera una pizarra en nuestra pantalla, y definíamos un tamaño de 600 de ancho por 400 de altura. Aquí habíamos creado una variable que se llama pantalla, que es nuestro conector utilizando querySelector entre JavaScript y HTML.

[02:02] Creamos una variable pincel que le decimos que nos traiga todo el contenido de pantalla, getContext, en dos dimensiones. Estamos trabajando

con un largo y un ancho. Aquí habíamos creado un rectángulo. Esto de aquí voy a eliminar también, era una explicación de la anterior aula.

[02:32] Habíamos creado un rectángulo plomo de 600 x 400, básicamente el tamaño de nuestra pizarra. Lo que vamos a hacer ahora es crear un cuadrado de 50 x 50. Voy a guardar. Voy a abrir aquí una nueva página de un nuevo Chrome. Aquí lo tenemos, vamos a hacer lo mismo que ya teníamos para organizar nuestro código, y aquí voy a abrir ese programita. Aquí estamos.

[03:17] Entonces, tengo mi cuadrado de 50 x 50. Voy ahora a graficar otro cuadrado aquí al lado. Entonces lo único que tengo que hacer es moverme de posición. Entonces, le estoy diciendo: "Andate a la posición 50, 0 que es esta de aquí, y graficame otro cuadrado". Voy a guardar y actualizamos.

[03:50] Está aquí el otro cuadrado, solo que como tiene las mismas características pareciera que fuera un rectángulo. Para evitar eso, vamos a graficarle un borde a cada uno de nuestros cuadrados. Para eso, vamos a usar una función dentro del mismo pincel donde vamos a definir el estilo de línea, `strokeStyle`, y lo vamos a definir color negro, en inglés `black`.

[04:34] Y aquí vamos a dibujar nuestro cuadrado en este caso, nuestro borde, que sería `strokeRect`, de rectángulo, y tendría que tener los mismos parámetros de lo que estamos pintando. Entonces, esto de aquí es como si pintáramos con una brocha o con un rodillo, y el `stroke` es como si dibujáramos con una regla alrededor de nuestro cuadrado.

[04:52] Entonces, para probar si está correcto aquí está. Entonces, el primer cuadradito de 50 x 50 está con borde y el otro no. Le vamos a hacer borde a los dos. Aquí me está faltando punto y coma. Le hacemos un borde a los dos. Perfecto. Guardo y actualizo. Aquí nos habíamos olvidado, no olvidemos, tienen que tener las mismas coordenadas. Aquí básicamente había graficado en el mismo lugar.

[05:56] Esto de aquí, quiero licencia, no. Entonces, ya los dos tienen.

Imagínense que quiero otro cuadrado. En realidad aquí no necesitaba volver a definir los colores. Eso solo defino una vez, si es que no voy a cambiar, y ahora, imagínense que quiero otro cuadrado.

[06:19] Aquí tengo que volverle a decir, ya no es en la posición 50, sino en la posición 100. Si quiero cinco cuadrados aquí va a ser en la posición 150, 150, en la posición 200, en la posición 200. Y tengo mis cinco cuadrados. Ahora, ¿qué tal si me piden 1000 cuadrados? ¿Les recuerda a algo este tipo de problemas?

[06:54] Ya nos hemos topado, hemos enfrentado este tipo de problemas en las anteriores etapas del curso, donde lo que hacíamos era utilizar, reutilizar funciones para poder optimizar código, para poder reutilizar y ser mucho más rápidos a la hora de construir nuestro programa. Porque aquí, si tengo que construir mil cuadrados, tengo que escribir mil veces nuestro código.

[07:24] Entonces, lo que vamos a hacer es crear una función, voy a eliminar todo esto, voy a crear una función que la vamos a llamar function dibujarCuadradoVerde, va a ser bien específica, y aquí adentro voy a colocar todo este código, que es lo que nuestra función hace.

[08:00] Entonces, mi función está creada, voy a guardar, actualizo aquí. No tiene nada porque no la estoy llamando, entonces, lo único que el código hizo hasta ahora fue cargar función. Ahí está, esperando a ser llamada. Vamos a llamar nuestra función. DibujarCuadradoVerde y vamos a llamarla cuatro veces, cuatro cuadrados.

[08:31] Tenemos algún problema. Como se habrán dado cuenta, no estoy pasando aquí ningún parámetro, estoy graficando el cuadradito verde de 50 x 50 siempre en la misma posición, entonces, aquí lo grafico, en la segunda estoy sobreponiendo en el cuadrado que ya tenía, en el tercero estoy sobreponiendo en el cuadrado que ya tenía y en el cuarto estoy volviendo a sobreponer.

[09:01] Entonces, mi función no está atendiendo la necesidad inicial que yo tenía, que era dibujar varios cuadrados aquí en nuestra pizarra de 600 x 400, en nuestro Canvas. Entonces, vamos a ver en la próxima sección cómo solucionamos nuestro problema con nuestra función.